

# Repetition-Free Derivability from a Regular Grammar is NP-Hard

Jochen Burghardt

*jochen.burghardt@alumni.tu-berlin.de*

February 2016

---

## Abstract

We prove the NP-hardness of the problem whether a given word can be derived from a given regular grammar without repeated occurrence of any nonterminal.

*Key words:* Regular grammar; repetition-free derivation; NP-hard

---

## 1 Introduction

Let a regular word grammar  $\mathcal{G}$  be given. We ask whether a given word  $\omega$  can be derived from  $\mathcal{G}$  without repeated occurrence of any nonterminal. We prove in Sect. 3 that the problem of deciding this property is NP-hard in general. As a consequence, it is NP-hard also for all superclasses of regular grammars, such as context-free, context-sensitive, and unrestricted grammars.

In Sect. 4, we present some ideas to prove the NP-hardness of a related problem, viz. of determining the length of the longest word repetition-free derivable from a given grammar. However, we didn't yet succeed in finding a proof for that claim.

In Sect. 5, we present the original motivation of considering repetition-free derivations, which was a rather particular problem from artificial intelligence.

The problem of deciding repetition-free derivability looks quite similar to that of deciding the existence of a Hamiltonian path in a given undirected graph, which is well-known to be NP-complete [Sip97, Thm.7.35, Sect.7.5, p.262]. However, both problems differ in

- presence of terminals/edge labels,
- the set of nonterminals/nodes in a derivation/path (arbitrary vs. full set), and
- the admitted start and end nonterminals/nodes of a derivation/path (fixed start and end symbols vs. arbitrary nodes), respectively.

For this reason, a reduction of the Hamiltonian path problem to the repetition-free derivability problem is not immediate obvious.

## 2 Definitions

**Definition 1** (*Regular grammar*) Following [HU79, Sect.9.1/4.2, p.217/79], a regular (word) grammar  $\mathcal{G}$  is defined as a tuple  $\langle \mathcal{N}, \Sigma, \mathcal{R}, S \rangle$ , where  $\mathcal{N}$  and  $\Sigma$  are disjoint finite sets of nonterminal and terminal symbols, respectively,  $S \in \mathcal{N}$  is called the start symbol, and  $\mathcal{R}$  is a finite set of rules of the form  $A ::= bC$  or  $A ::= b$ , where  $A, C \in \mathcal{N}$  and  $b \in \Sigma$ .

A derivation from  $\mathcal{G}$  is a finite sequence

$$\begin{aligned} S &\rightarrow a_1 X_1 \\ &\rightarrow a_1 a_2 X_2 \\ &\rightarrow \dots \\ &\rightarrow a_1 a_2 \dots a_{n-1} X_{n-1} \\ &\rightarrow a_1 a_2 \dots a_{n-1} a_n X_n \\ &\rightarrow a_1 a_2 \dots a_{n-1} a_n a_{n+1} \end{aligned}$$

where  $a_1, \dots, a_{n+1} \in \Sigma$  are terminal symbols,  $X_1, \dots, X_n \in \mathcal{N}$  are nonterminal symbols, and

$$\begin{aligned} S &::= a_1 X_1, \\ X_1 &::= a_2 X_2, \\ \dots &, \\ X_{n-1} &::= a_n X_n, \text{ and} \\ X_n &::= a_{n+1} \end{aligned}$$

are rules from  $\mathcal{R}$ . We say that the nonterminals  $X_1, \dots, X_n$  occur in that derivation. A word  $\omega \in \Sigma^*$  is derivable from  $\mathcal{G}$  if a derivation  $S \rightarrow \dots \rightarrow \omega$  exists. The language produced by  $\mathcal{G}$  is denoted by  $\mathcal{L}(\mathcal{G})$ , it is defined as the set of all words derivable from  $\mathcal{G}$ .  $\square$

**Definition 2** (*Conjunctive normal form formula*) Let a set  $\{x_1, \dots, x_m\}$  of propositional variables be given. A boolean formula in (3-literal) conjunctive normal form is given as a conjunction  $\kappa = \kappa_1 \cdot \dots \cdot \kappa_n$ , where the  $j$ .th conjunct  $\kappa_j$  has the form  $y_{j1} + y_{j2} + y_{j3}$  and each literal  $y_{jk}$  satisfies  $y_{jk} \in \{x_1, \dots, x_m\} \cup \{\bar{x}_1, \dots, \bar{x}_m\}$ .

Given an assignment of truth values 0 or 1 to the variables  $x_1, \dots, x_m$ ,

- a literal  $x_i$  and  $\bar{x}_i$  is satisfied if 1 and 0 has been assigned to  $x_i$ , respectively;
- a conjunct  $\kappa_j = y_{j1} + y_{j2} + y_{j3}$  is satisfied if at least one of its literals  $y_{j1}, y_{j2}, y_{j3}$  is; and
- the whole formula  $\kappa = \kappa_1 \cdot \dots \cdot \kappa_n$  is satisfied if each of its conjuncts  $\kappa_j$  is.

The formula is called satisfiable if it is satisfied by some assignment. It is well-known that the problem of deciding the satisfiability of a given 3-literal conjunctive normal form formula is NP-complete (e.g. [AHU74, Sect.10.4, Thm.10.4, p.384]).  $\square$

---

|                                      |                                               |
|--------------------------------------|-----------------------------------------------|
| $S_{i-1} ::= a X_{i1}$               | for $i = 1, \dots, m$                         |
| $S_{i-1} ::= a \bar{X}_{i1}$         | for $i = 1, \dots, m$                         |
| $X_{ij} ::= a X_{i,j+1}$             | for $i = 1, \dots, m$ and $j = 1, \dots, n-1$ |
| $\bar{X}_{ij} ::= a \bar{X}_{i,j+1}$ | for $i = 1, \dots, m$ and $j = 1, \dots, n-1$ |
| $X_{in} ::= a S_i$                   | for $i = 1, \dots, m$                         |
| $\bar{X}_{in} ::= a S_i$             | for $i = 1, \dots, m$                         |
| $S_m ::= bT_0$                       |                                               |

---

|                             |                                         |
|-----------------------------|-----------------------------------------|
| $T_{j-1} ::= c \gamma_{jk}$ | for $j = 1, \dots, n$ and $k = 1, 2, 3$ |
| $\gamma_{jk} ::= e T_j$     | for $j = 1, \dots, n$ and $k = 1, 2, 3$ |
| $T_n ::= d$                 |                                         |

---

where the mapping  $\gamma$  is defined by

|                              |                          |
|------------------------------|--------------------------|
| $\gamma_{jk} = X_{ij}$       | for $y_{jk} = x_i$       |
| $\gamma_{jk} = \bar{X}_{ij}$ | for $y_{jk} = \bar{x}_i$ |

---

Fig. 1. Grammar rules in Def. 3

### 3 Repetition-free derivability

The ordinary derivability problem for regular word grammars can be solved within an time upper bound of  $\mathcal{O}(n \cdot s^2)$ , where  $n$  and  $s$  is the length of the input string and the number of nonterminals, respectively [HMU03, Sect.4.3.3, p.153].<sup>1</sup> In contrast, repetition-free derivability is NP-hard, as we show in the following.

We reduce the satisfiability problem for conjunctive normal forms, which is well-known to be NP-complete [AHU74, Thm.10.3, Sect.10.4, p.379], to the repetition-free derivability problem. We give the mapping of a former to a latter problem in Def. 3, and prove it a reduction in Cor. 6, based essentially on Lem. 5.

**Definition 3** (*Grammar corresponding to a conjunctive normal form*) Given a conjunctive normal form formula as in Def. 2, we define a “corresponding” a regular grammar  $\mathcal{G} = \langle \mathcal{N}, \Sigma, \mathcal{R}, S_0 \rangle$  as follows.

Let  $\mathcal{N} = \{S_0, \dots, S_m, T_0, \dots, T_n\} \cup \{X_{ij}, \bar{X}_{ij} \mid 1 \leq i \leq m \wedge 1 \leq j \leq n\}$  be the set of nonterminal symbols, let  $\Sigma = \{a, b, c, d\}$  be the set of terminal symbols. Let the rules  $\mathcal{R}$  be as shown in Fig 1. We refer to the topmost 7 and the next 3 lines as the upper and lower grammar part, respectively.  $\square$

**Example 4** (*Conjunctive normal form and corresponding grammar*) As an example, the conjunctive normal form

$$(x_1 + \bar{x}_2 + x_4) \cdot (x_2 + x_3 + \bar{x}_4) \cdot (\bar{x}_1 + \bar{x}_2 + x_4)$$

<sup>1</sup> Hopcroft et. al. explain their algorithm on nondeterministic finite automata, using the number of states for  $s$ . However, carrying-over to regular grammars is straight-forward.

|                                      |                                            |
|--------------------------------------|--------------------------------------------|
| $S_0 ::= aX_{11} \mid a\bar{X}_{11}$ |                                            |
| $X_{11} ::= aX_{12} \mid eT_1$       | $\bar{X}_{11} ::= a\bar{X}_{12}$           |
| $X_{12} ::= aX_{13}$                 | $\bar{X}_{12} ::= a\bar{X}_{13}$           |
| $X_{13} ::= aS_1$                    | $\bar{X}_{13} ::= aS_1 \mid eT_3$          |
| $S_1 ::= aX_{21} \mid a\bar{X}_{21}$ |                                            |
| $X_{21} ::= aX_{22}$                 | $\bar{X}_{21} ::= a\bar{X}_{22} \mid eT_1$ |
| $X_{22} ::= aX_{23} \mid eT_2$       | $\bar{X}_{22} ::= a\bar{X}_{23}$           |
| $X_{23} ::= aS_2$                    | $\bar{X}_{23} ::= aS_2 \mid eT_3$          |
| $S_2 ::= aX_{31} \mid a\bar{X}_{31}$ |                                            |
| $X_{31} ::= aX_{32}$                 | $\bar{X}_{31} ::= a\bar{X}_{32}$           |
| $X_{32} ::= aX_{33} \mid eT_2$       | $\bar{X}_{32} ::= a\bar{X}_{33}$           |
| $X_{33} ::= aS_3$                    | $\bar{X}_{33} ::= aS_3$                    |
| $S_3 ::= aX_{41} \mid a\bar{X}_{41}$ |                                            |
| $X_{41} ::= aX_{42} \mid eT_1$       | $\bar{X}_{41} ::= a\bar{X}_{42}$           |
| $X_{42} ::= aX_{43}$                 | $\bar{X}_{42} ::= a\bar{X}_{43} \mid eT_2$ |
| $X_{43} ::= aS_4 \mid eT_3$          | $\bar{X}_{43} ::= aS_4$                    |
| $S_4 ::= bT_0$                       |                                            |

$$\begin{aligned}
T_0 &::= cX_{11} \mid c\bar{X}_{21} \mid cX_{41} \\
T_1 &::= cX_{22} \mid cX_{32} \mid c\bar{X}_{42} \\
T_2 &::= c\bar{X}_{13} \mid c\bar{X}_{23} \mid cX_{43} \\
T_3 &::= d
\end{aligned}$$

Fig. 2. Example grammar in Exm. 4

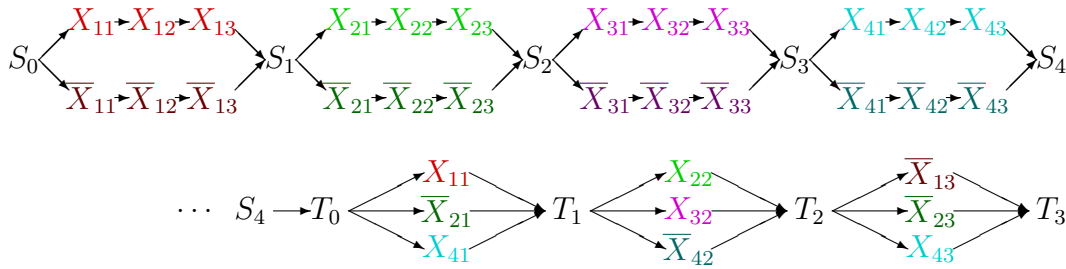


Fig. 3. Illustration of example grammar in Exm. 4

corresponds to the grammar shown in Fig. 2, where different colors indicate different variables, while light and dark shades indicate unnegated and negated occurrences, respectively. The  $S_j ::= \dots$  rules of the lower part are shown bottom right, its  $\gamma_{j+1,k} ::= \dots$  are integrated as alternatives in the upper part's rules. See also the illustration in Fig. 3, where upper and lower part are strictly separated, and their common nonterminals (like  $X_{11}$ ) are shown twice. Observe that no nonterminal occurs multiply in the upper part alone, and likewise none does in the lower.  $\square$

| $y_{jk}$    | $x_i$ | sat | initial        | final          | rep |
|-------------|-------|-----|----------------|----------------|-----|
| $x_i$       | 0     | −   | $X_{ij}$       | $X_{ij}$       | +   |
| $\bar{x}_i$ | 0     | +   | $X_{ij}$       | $\bar{X}_{ij}$ | −   |
| $x_i$       | 1     | +   | $\bar{X}_{ij}$ | $X_{ij}$       | −   |
| $\bar{x}_i$ | 1     | −   | $\bar{X}_{ij}$ | $\bar{X}_{ij}$ | +   |

Fig. 4. Satisfied literal vs. repetition-free  $ce$  derivation in Lem. 6

**Lemma 5** (*Repetition-Free derivability*) *Given a conjunctive normal form formula  $\kappa$  as in Def. 2, and its corresponding grammar  $\mathcal{G}$  as in Def. 3, the word  $\omega = a^{(n+1) \cdot m} b (ce)^n d$  has a repetition-free derivation from  $\mathcal{G}$  iff  $\kappa$  has a satisfying variable assignment.*

**PROOF.** First, note that symbols  $a$  and  $b$  are only produced by the upper grammar part; similarly, symbols  $c$  and  $d$  are only produced by the lower one. Therefore, in order to derive a word starting with  $a^{(n+1) \cdot m} b$ , the rules of the upper grammar part must be applied  $(n+1) \cdot m + 1$  times, leading to an initial derivation part  $S_0 \xrightarrow{*} a^{(n+1) \cdot m} S_m \rightarrow a^{(n+1) \cdot m} b T_0$ . Similarly, a word ending in  $(ce)^n d$  can be derived only by applying the lower part rules  $2 \cdot n + 1$  times, leading to a final derivation part  $T_0 \xrightarrow{*} (ce)^n T_n \rightarrow (ce)^n d$ . Hence, each derivation of  $\omega$  from  $\mathcal{G}$  can be decomposed into an initial and a final part with those properties.

Next, observe that the transitive closure of the relation  $\succ$  on  $\mathcal{N}$ , defined by

$$A \succ B \quad \text{if} \quad A ::= zB \text{ is an upper part rule for some } z \in \Sigma,$$

is asymmetric, i.e. an ordering relation. Therefore, a part of a derivation of  $\omega$  from  $\mathcal{G}$  that uses only rules from  $\mathcal{G}$ 's upper part cannot have any nonterminal repetition. For a similar reason, no derivation part using only rules from the lower part can have any nonterminal repetition. Hence, the only way a nonterminal repetition can occur in a derivation of  $\omega$  is to repeat a nonterminal from the initial derivation part in the final part.

There are  $2^m$  different initial derivation parts  $S_0 \xrightarrow{*} a^{(n+1) \cdot m} b T_0$ . For each  $i = 1, \dots, m$ , either all of  $\bar{X}_{i1}, \dots, \bar{X}_{in}$  but none of  $X_{i1}, \dots, X_{in}$  occur in an initial derivation part, or vice versa. Each assignment of the variables  $x_1, \dots, x_m$  corresponds uniquely to an initial derivation part such that  $x_i$  is assigned 1 iff  $\bar{X}_{ij}$  occurs in the part but  $X_{ij}$  does not, for  $j = 1, \dots, n$ .

Assume some fixed initial derivation part  $S_0 \xrightarrow{*} a^{(n+1) \cdot m} b T_0$  has been chosen, corresponding to some fixed truth value assignment to  $x_1, \dots, x_m$ . As Fig. 4 demonstrates, a subsequent derivation  $T_{j-1} \rightarrow c\gamma_{jk} \rightarrow ceT_j$  causes a repetition iff the literal  $y_{ik}$  in the  $j$ th conjunct isn't true in the chosen assignment:

- Column  $y_{jk}$  lists the possible forms that this literal can take, where  $i$  is chosen such that  $y_{jk} \in \{x_i, \bar{x}_i\}$ ,
- column  $x_i$  lists the possible truth values assigned to  $x_i$ ,
- column “sat” shows for each possibility whether the literal  $y_{jk}$  is satisfied (“+”) or not (“−”),

- column “initial” shows, for each possibility, the nonterminal of the initial derivation part corresponding to the assignment to  $x_i$ ,
- column “final” shows, for each possibility, the nonterminal  $\gamma_{jk}$  of the final derivation part  $T_{j-1} \rightarrow c\gamma_{jk} \rightarrow ceT_j$ ,
- column “rep” shows, for each possibility, whether the latter nonterminal of the final part is a repetition of that from the initial part.

Since each possible path  $T_{j-1} \rightarrow \rightarrow ceT_j$  involves some  $\gamma_{jk}$ , each such path causes a nonterminal repetition iff the  $j$ th conjunct,  $y_{j1} + y_{j2} + y_{j3}$ , isn't satisfied by the assignment.

Since the only way to have a repetition is between the initial part and some  $T_{j-1} \rightarrow \rightarrow ceT_j$  part, we have: Each derivation of  $\omega$  starting with the chosen initial derivation part leads to a repetition iff the corresponding truth value assignment doesn't satisfy the formula.

Hence, no repetition-free derivation of  $\omega$  exists iff the formula is unsatisfiable.  $\square$

**Corollary 6** (*Repetition-Free Derivability from a Regular Grammar is NP-Hard*)  
*The task to decide whether a given word  $\omega$  has a derivation without nonterminal repetition from a given regular grammar  $\mathcal{G}$  is NP-hard.*

**PROOF.** Let a conjunctive normal form formula  $\kappa$  be given as in Def. 2. Let  $\mathcal{G}$  be the corresponding grammar as in Def. 3, let  $\omega = a^{(n+1) \cdot m} b (ce)^n d$ . By Lem. 5, the NP-complete problem to decide whether  $\kappa$  is satisfiable can be reduced to the task to decide whether  $\omega$  is derivable from  $\mathcal{G}$  without nonterminal repetition.  $\square$

**Example 7** (*Satisfiability and repetition-free derivability*) *Continuing Exm. 4, we consider derivations of the word  $\omega = a^{16}b(ce)^3d$ ; this word is derivable in a large number of ways. Each derivation contains an initial segment like e.g.*

$$\begin{aligned}
S_0 &\rightarrow a \textcolor{red}{X}_{11} \rightarrow aa \textcolor{red}{X}_{12} \rightarrow a^3 \textcolor{red}{X}_{13} \rightarrow a^4 S_1 \\
&\rightarrow a^5 \textcolor{green}{\overline{X}}_{21} \rightarrow a^6 \textcolor{green}{\overline{X}}_{22} \rightarrow a^7 \textcolor{green}{\overline{X}}_{23} \rightarrow a^8 S_2 \\
&\rightarrow a^9 \textcolor{violet}{X}_{31} \rightarrow a^{10} \textcolor{violet}{X}_{32} \rightarrow a^{11} \textcolor{violet}{X}_{33} \rightarrow a^{12} S_3 \\
&\rightarrow a^{13} \textcolor{teal}{\overline{X}}_{41} \rightarrow a^{14} \textcolor{teal}{\overline{X}}_{42} \rightarrow a^{15} \textcolor{teal}{\overline{X}}_{43} \rightarrow a^{16} S_4 \rightarrow a^{16}b T_0,
\end{aligned}$$

where for each variable  $x_i$  either all nonterminals  $X_{i1}, X_{i2}, X_{i3}$ , or all nonterminals  $\overline{X}_{i1}, \overline{X}_{i2}, \overline{X}_{i3}$  occur; this corresponds to an assignment of 0 or 1 to  $x_i$ . In our initial segment example, the derivation corresponds to the assignment  $x_1 = x_3 = 0$  and  $x_2 = x_4 = 1$ . In a final segment, we have derivations like

$$T_0 \rightarrow c\textcolor{red}{X}_{11} \rightarrow ceT_1 \rightarrow cec\textcolor{green}{X}_{22} \rightarrow (ce)^2T_2 \rightarrow (ce)^2c\textcolor{teal}{X}_{43} \rightarrow (ce)^3T_3 \rightarrow (ce)^3b.$$

Such a derivation may contain a repetition of a nonterminal from the initial segment. In our example,  $T_0 \rightarrow cX_{11} \rightarrow ceT_1$  contains the repetition of  $X_{11}$ , and correspondingly the propositional variable occurrence  $x_1$  in the first conjunct is not satisfied by the above assignment. However,  $T_0 \rightarrow cX_{41} \rightarrow ceT_1$  does not contain a repetition, and the first conjunct is satisfied by the assignment since  $x_4$  is.  $\square$

#### 4 Longest repetition-free derivable words

We suspect that the correspondance from Def. 3 between formula  $\kappa$  and grammar  $\mathcal{G}$ , or a slightly modified version, can also be used to prove NP-hardness of the problem of determining the length of the longest word derivable from a given grammar without repetition.

We already achieved, in Lem. 8, to establish that no word longer than  $\omega$  from Lem. 5, i.e. longer than  $(n + 1) \cdot (m + 2)$  symbols, can be derived repetition-free from  $\mathcal{G}$ .

If  $\omega$  was the only word of its length that was repetition-free derivable from  $\mathcal{G}$ , we had that the longest repetition-free derivable word has length  $(n + 1) \cdot (m + 2)$  iff  $\kappa$  is satisfiable, and a properly shorter length otherwise. However, as Exm. 9 shows, there are other words of length  $(n + 1) \cdot (m + 2)$  that are repetition-free derivable from  $\mathcal{G}$ , but don't correspond to a truth value assignment in an obvious way. If we always could construct from such a word a corresponding satisfying assignment, we had proven the suspected NP-hardness result.

**Lemma 8** (*Upper bound for repetition-free derivable words*) *No word longer than  $(n + 1) \cdot (m + 2)$  can be derived repetition-free from the grammar  $\mathcal{G}$  from Def. 3.*

**PROOF.** Let  $\psi$  be a word that can be derived repetition-free from  $\mathcal{G}$ . First,  $\psi$  contains exactly one symbol  $d$ . Next, every production of a symbol  $b$  or  $e$  increases the number of nonterminals from  $\{T_0, \dots, T_n\}$  that occurred in the derivation, hence  $\psi$  can contain at most  $n + 1$  such symbols.

We now prove an upper bound on the total number of  $a$  and  $c$  symbols in  $\psi$ . Assign a pair  $\langle s^*, j^* \rangle$  to every intermediate word in the derivation chain of  $\psi$ , where

- $s^*$  is the number of nonterminals from  $\{S_0, \dots, S_m\}$  that already occurred, and
- $j^*$  is the current “conjunction index”, i.e.
  - $j^* = j$  if the current nonterminal is  $T_j$  or some  $X_{ij}$  or  $\bar{X}_{ij}$ ,
  - $j^* = 0$  if the current nonterminal is some  $S_i$ , and
  - $j^* = n$  if the current word doesn't contain a nonterminal.

We inspect the grammar rules from Fig. 1 to show that the current pair is properly increased wrt. the lexicographical order whenever a symbol  $a$  or  $c$  is produced:

- If  $S_{i-1} ::= aX_{i1}$  or  $S_{i-1} ::= a\bar{X}_{i1}$  is applied,  $s^*$  remains unchanged, while  $j^*$  is increased from 0 to 1.
- If or  $X_{ij} ::= aX_{i,j+1}$  or  $\bar{X}_{ij} ::= a\bar{X}_{i,j+1}$  is applied,  $s^*$  remains unchanged, while  $j^*$  is increased from  $j$  to  $j + 1$ .
- If  $X_{in} ::= aS_i$  or  $\bar{X}_{in} ::= aS_i$  is applied,  $s^*$  is increased, while  $j^*$  is reset to 0.

- If  $T_{j-1} ::= c\gamma_{jk}$  is applied for some  $k \in \{1, 2, 3\}$ ,  
 $s^*$  remains unchanged, while  $j^*$  is increased from  $j - 1$  to  $j$ .

The remaining rules don't modify the current pair:

- If  $S_m ::= bT_0$  is applied,  $s^*$  remains unchanged, and  $j^*$  remains 0.
- If  $\gamma_{jk} ::= eT_j$  is applied for some  $k \in \{1, 2, 3\}$ ,  
 $s^*$  remains unchanged, and  $j^*$  remains  $j$ .
- If  $T_n ::= d$  is applied,  $s^*$  remains unchanged, and  $j^*$  remains  $n$ .

Since  $S_0$  occurs in every intermediate word, we have  $1 \leq s^* \leq m+1$  and  $0 \leq j^* \leq n$  for every possible pair  $\langle s^*, j^* \rangle$ . Hence, there are  $(m+1) \cdot (n+1)$  possible pairs, and the current pair can be increased at most  $(m+1) \cdot (n+1) - 1$  times. Therefore, there are at most that much  $a$  and  $c$  occurrences in  $\psi$ .

Summing up, the length of  $\psi$  cannot exceed  $1 + n + 1 + (m+1) \cdot (n+1) - 1 = (m+2) \cdot (n+1)$  symbols.  $\square$

**Example 9** (*Length issues*) Continuing Exm. 4 and 7, observe that there are repetition-free derivable words of length  $(m+2) \cdot (n+1)$  that are different from  $\omega$  and don't correspond to a truth value assignment. An examples is

$$\begin{aligned}
S_0 &\rightarrow a \textcolor{red}{X}_{11} \xrightarrow{*} a^5 \textcolor{green}{X}_{21} \xrightarrow{*} a^9 \textcolor{violet}{X}_{31} \xrightarrow{*} a^{13} \textcolor{teal}{X}_{41} \\
&\rightarrow a^{13}e T_1 \rightarrow a^{13}ec \textcolor{teal}{\overline{X}}_{42} \rightarrow a^{13}ece T_2 \rightarrow a^{13}ecec \textcolor{teal}{X}_{43} \\
&\rightarrow a^{13}ececa S_4 \rightarrow a^{13}ececab T_0 \rightarrow a^{13}ececabc \textcolor{teal}{\overline{X}}_{21} \\
&\rightarrow a^{13}ececabca \textcolor{teal}{\overline{X}}_{22} \rightarrow a^{13}ececabcaa \textcolor{teal}{\overline{X}}_{23} \\
&\rightarrow a^{13}ececabcaae T_3 \rightarrow a^{13}ececabcaaed
\end{aligned}$$

This derivation cannot correspond to a variable assignment, since it contains e.g. both  $\textcolor{green}{X}_{21}$  and  $\textcolor{teal}{\overline{X}}_{21}$ . By Lem. 8, no longer word can be derived from the example grammar.

As a side remark, there are shorter words derivable from  $S_0$  without repetition, such as

$$S_0 \rightarrow a\textcolor{red}{X}_{11} \rightarrow aeT_1 \rightarrow aec\textcolor{green}{X}_{22} \rightarrow aeceT_2 \rightarrow aecec\textcolor{red}{\overline{X}}_{13} \rightarrow ae(ce)^2T_3 \rightarrow ae(ce)^2d$$

and

$$S_0 \rightarrow a\textcolor{red}{\overline{X}}_{11} \rightarrow aa\textcolor{red}{\overline{X}}_{12} \rightarrow a^3\textcolor{red}{\overline{X}}_{13} \rightarrow a^4eT_3 \rightarrow a^4ed.$$

Note that the former derivation also no longer corresponds to a variable assignment, since it contains both  $\textcolor{red}{X}_{11}$  and  $\textcolor{red}{\overline{X}}_{13}$ . When repetitions are allowed, arbitrarily long words can be derived, e.g.

$$\begin{aligned}
S_0 &\xrightarrow{*} a^{16}b T_0 \\
&\rightarrow a^{16}bc \textcolor{teal}{X}_{41} \rightarrow a^{16}bc a \textcolor{teal}{X}_{42} \xrightarrow{*} a^{16}bc a^3 S_4 \xrightarrow{*} a^{16}bc a^3bc \textcolor{teal}{X}_{41} \\
&\xrightarrow{*} a^{16}bc (a^3bc)^r e(ce)^2d
\end{aligned}$$

for any  $r \geq 0$ .  $\square$



|                                      |                                             |
|--------------------------------------|---------------------------------------------|
| $S_{i-1} ::= a X_{in}$               | for $i = 1, \dots, m$                       |
| $S_{i-1} ::= a \bar{X}_{in}$         | for $i = 1, \dots, m$                       |
| $X_{ij} ::= a X_{i,j-1}$             | for $i = 1, \dots, m$ and $j = n, \dots, 2$ |
| $\bar{X}_{ij} ::= a \bar{X}_{i,j-1}$ | for $i = 1, \dots, m$ and $j = n, \dots, 2$ |
| $X_{i1} ::= a S_i$                   | for $i = 1, \dots, m$                       |
| $\bar{X}_{i1} ::= a S_i$             | for $i = 1, \dots, m$                       |
| $S_m ::= bT_0$                       |                                             |
| $T_{j-1} ::= c \gamma_{jk}$          | for $j = 1, \dots, n$ and $k = 1, 2, 3$     |
| $\gamma_{jk} ::= e T_j$              | for $j = 1, \dots, n$ and $k = 1, 2, 3$     |
| $T_n ::= d$                          |                                             |

where the mapping  $\gamma$  is defined by

|                              |                          |
|------------------------------|--------------------------|
| $\gamma_{jk} = X_{ij}$       | for $y_{jk} = x_i$       |
| $\gamma_{jk} = \bar{X}_{ij}$ | for $y_{jk} = \bar{x}_i$ |

Fig. 5. Reversed grammar rules in Def. 3

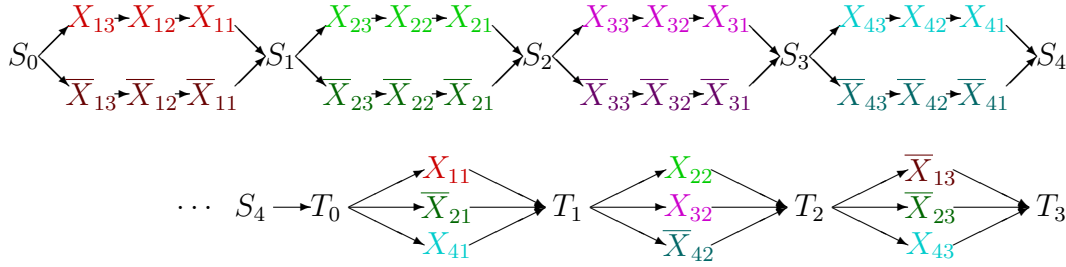


Fig. 6. Illustration of reversed example grammar in Exm. 4

In an attempt to remedy the above problems, we modified the grammar from Def. 3 as shown in Fig. 5. In the upper part, the  $X_{ij}$  are chained in reverse order, as are the  $\bar{X}_{ij}$ . The corresponding example grammar for Exm. 4 is illustrated in Fig. 6.

Almost similar to Lem. 8, we established a length upper bound of  $(n+1) \cdot (m+2)$  for repetition-free derivations from the reversed grammar, see Lem. 10. The requirement that a word contains a “b” symbol could possibly be overcome if the upper and the lower part were concatenated in reverse order, i.e. by deleting the rules  $S_m ::= bT_0$  and  $T_n ::= d$ , adding instead the rules  $S_m ::= d$  and  $T_n ::= bS_0$ , and changing the start symbol to be  $T_0$ . However, we didn’t elaborate this modification.

**Lemma 10** (*Upper bound for repetition-free derivable words (reversed grammar)*)  
For  $n \geq 2$ , no word longer than  $(n+1) \cdot (m+2)$  and containing a “b” symbol can be derived repetition-free from the grammar  $\mathcal{G}$  from Def. 3.

**PROOF.** Let  $\psi$  be a word that can be derived repetition-free from  $\mathcal{G}$ . Let  $a$ ,  $c$ , and  $e$  denote the number of occurrences of “a”, “c”, and “e” in  $\psi$ , respectively.

Assign a “conjunction index” to every nonterminal as follows:

- assign  $j$  to each  $X_{ij}$ , for  $i = 1, \dots, m$  and  $j = 1, \dots, n$ ,
- assign  $n + 1$  to each  $S_i$ , for  $i = 0, \dots, m$ , and
- assign  $j + 1$  to each  $T_j$ , for  $j = 0, \dots, n$ .

Observe the following properties:

- Each increase of the conjunction index in the derivation requires some  $S_i$  or  $T_j$  to occur; neither an occurrence of  $S_0$  nor one of  $T_0$  leads to an increase.
- More precisely, the conjunction index is increased from 1 to  $n + 1$  when some  $S_i$  occurs, and from  $j$  to  $j + 1$  when some  $T_j$  occurs.
- Hence, the conjunction index can experience at most a total increase of  $mn + n$ , if all  $m + n$  rules producing a  $S_i$  or  $T_j$  are used.
- Both the initial and the final conjunction index is  $n + 1$ .
- Hence the conjunction index’ total increase must equal the total decrease.
- If rule  $S_m ::= bT_0$  is applied, decreasing the conjunction index from  $n + 1$  to 1, at most  $mn$  “a”-producing rules can be applied, each of them decreasing the conjunction index by 1. That is, there are at most  $1 + mn$  decreasing rule applications.
- Each grammar rule changes the conjunction index, except where a “c” is produced, by a rule  $T_{j-1} ::= c\gamma_{jk}$ .
- Adding up the upper bound for the number of rule applications that increase, decrease, and keep the conjunction index, and the inevitable final one  $T_n ::= d$ , we get  $(m + n) + (mn + 1) + n + 1 = (n + 1)(m + 2)$ .  $\square$

**Example 11** (*Length issues (reversed grammar)*) For the reversed grammar scheme, there are still derivable words of length  $(m + 2) \cdot (n + 1)$  that are different from  $\omega$  and don’t correspond to a truth value assignment. An example, based on the grammar for  $(x_1 + \bar{x}_2 + x_4) \cdot (x_1 + x_3 + \bar{x}_1) \cdot (\bar{x}_1 + \bar{x}_2 + x_2)$  is the following.

$$\begin{aligned}
S_0 &\rightarrow a X_{13} \rightarrow aa X_{12} \rightarrow aae T_2 \\
&\rightarrow aaec \bar{X}_{23} \rightarrow aaeca \bar{X}_{22} \rightarrow aaecaa \bar{X}_{21} \\
&\rightarrow aaeca^3 S_2 \rightarrow aaeca^4 \bar{X}_{33} \xrightarrow{*} aaeca^8 \bar{X}_{43} \\
&\xrightarrow{*} aaeca^{11} S_4 \rightarrow aaeca^{11} b T_0 \rightarrow aaeca^{11} bc X_{11} \\
&\rightarrow aaeca^{11} bce T_1 \rightarrow aaeca^{11} bcec \bar{X}_{12} \rightarrow aaeca^{11} bceca \bar{X}_{11} \\
&\rightarrow aaeca^{11} bcecaa S_1 \rightarrow aaeca^{11} bceca^3 X_{23} \rightarrow aaeca^{11} bceca^3 e T_3 \\
&\rightarrow aaeca^{11} bceca^3 ed
\end{aligned}$$

Note that the 2nd and 3rd conjunct of the conjunctive normal form are trivial, as they contain a variable and its negation. It is not yet clear whether there are similar counter-examples for non-trivial normal forms.  $\square$

## 5 Application to sequence guessing

A modification of Cor. 6 can be applied to a problem in artificial intelligence; this was our original motivation to investigate repetition-free derivations.

One of the typical tasks in classical intelligence tests is to guess a plausible construction law for a given sequence of values. For example, the sequence 0; 2, 4, 6, 8 has construction laws like  $v_p * 2$  and  $v_1 + 2$ , where  $v_p$  and  $v_1$  denotes the position<sup>2</sup> within the sequence and the previous sequence value,<sup>3</sup> respectively.

Given a sequence  $s$  and a set  $\Sigma$  of admitted arithmetic operations, the set of all construction law terms for  $s$  that can be built from  $\Sigma$  can be computed as a regular tree grammar<sup>4</sup> by  $E$ -generalization<sup>5</sup> [Hei95], [Bur05, Sect.5.2, p.28–29].

As a formalization of Occam’s Razor, a law term should be as small as possible w.r.t. some user-definable notion of *size*; we call such a term *guessable* from the sequence. For any reasonable notion of size, a law term<sup>6</sup> should be discarded if a proper subterm constructs the same sequence, too. In the grammar setting, the latter condition amounts to discarding each term whose derivation uses a nonterminal repeatedly on the same term path. This is where repetition-free derivations come into play.

Based on our formalization, one may investigate various properties of a given intelligence test. Given  $\Sigma$ , a sequence  $s$ , and a proper prefix sequence  $s'$ , one may e.g. ask whether some law term  $t$  for  $s$  is guessable already from  $s'$ .<sup>7</sup> Since the law term grammar for  $s'$  is a quotient of the grammar  $\mathcal{G}$  for  $s$ , w.r.t. some equivalence relation  $\equiv$ , we are searching for a term  $t$  whose derivation from  $\mathcal{G}$  has no repetitions w.r.t.  $\equiv$ .

Corollary 14 below shows that this search task unfortunately is NP-hard already for the special case of regular word grammars.<sup>8</sup> It uses the technical result from Lem. 5.

Before giving the Corollary, we formalize some of the notions introduced above.

**Definition 12** (*Repetition-free derivation modulo equivalence*) *Given a regular grammar  $\mathcal{G}'$  and an equivalence relation  $\equiv$  on its set  $\mathcal{N}'$  of its nonterminals, define a derivation from  $\mathcal{G}'$  to be repetition-free mod.  $\equiv$  if it doesn’t contain two nonterminals that are equivalent mod.  $\equiv$ .  $\square$*

<sup>2</sup> starting with 0

<sup>3</sup> Since  $v_1$  is undefined at position 0, the first value cannot be constructed that way. We indicate by a semi-colon the first sequence position where a construction law shall apply.

<sup>4</sup> an extension of regular word grammars that share their closure and decidability properties, while describing sets of trees (i.e. terms), rather than words; their terminal symbols are function symbols of arbitrary arity; see e.g. [CDG<sup>+</sup>08]

<sup>5</sup> i.e. anti-unification w.r.t. an equational background theory defining the semantics of operations in  $\Sigma$

<sup>6</sup> e.g. (if  $v_p < 5$  then  $v_p * 2$  else 9) for the above example sequence

<sup>7</sup> In that case, being asked for a plausible continuation of  $s'$ , a valid answer would be  $s$ , based on the construction law  $t$  as a rationale. As a counter-example, the term (if  $v_p < 5$  then  $v_p * 2$  else 9) is guessable from 0, 2, 4, 6, 8, 9, but from none of its proper prefixes, since the subterm  $v_p * 2$  constructs each of them.

<sup>8</sup> i.e. even when all involved operator symbols are unary or nullary

**Definition 13** (*Quotient grammar*) Let  $\mathcal{G}' = \langle \mathcal{N}', \Sigma', \mathcal{R}', S' \rangle$  be a regular grammar, and  $\equiv$  be an equivalence relation on  $\mathcal{N}'$ . Similar to the construction of a quotient of a finite automaton,<sup>9</sup> we can define the quotient grammar  $\mathcal{G} = \mathcal{G}' / \equiv$  of  $\mathcal{G}'$  by  $\equiv$  to be  $\mathcal{G} = \langle \mathcal{N}, \Sigma, \mathcal{R}, S \rangle$ , where

- the nonterminal alphabet  $\mathcal{N} = \mathcal{N}' / \equiv$  of  $\mathcal{G}$  is the set of all equivalence classes of nonterminals from  $\mathcal{N}'$ ,
- the terminal alphabet  $\Sigma = \Sigma'$  of  $\mathcal{G}$  is shared with  $\mathcal{G}'$ ,
- the rules  $\mathcal{R}$  of  $\mathcal{G}$  are obtained by replacing all nonterminals in all rules in  $\mathcal{R}'$  by their equivalence classes, and
- the start symbol  $S = S' / \equiv$  of  $\mathcal{G}$  is the equivalence class of the start symbol of  $\mathcal{G}'$ .

It is obvious that every derivation from  $\mathcal{G}'$  can be “lifted” to a derivation from  $\mathcal{G}$ , by replacing each nonterminal by its equivalence class. Hence,  $\mathcal{L}(\mathcal{G}') \subseteq \mathcal{L}(\mathcal{G})$ , similar to the well-known property for quotient automata.  $\square$

**Corollary 14** (*Existence of repetition-free derivations mod. equivalence is NP-hard*) Given a regular grammar  $\mathcal{G}'$  and an equivalence relation  $\equiv$  on the set of its nonterminals, the problem to decide whether some word  $\omega \in \mathcal{L}(\mathcal{G}')$  has a derivation from  $\mathcal{G}'$  without repetitions mod.  $\equiv$ , is NP-hard in general.

**PROOF.** Let a conjunctive normal form formula  $\kappa$  be given as in Def. 2.

We construct a regular grammar  $\mathcal{G}'$  and an equivalence relation  $\equiv$  on its set  $\mathcal{N}'$  of nonterminal symbols such that: a word  $\omega \in \mathcal{L}(\mathcal{G}')$  exists that has a repetition-free derivation mod.  $\equiv$  iff  $\kappa$  has a satisfying variable assignment.

Let  $\mathcal{N}' = \{S_0, \dots, S_m, T_0, \dots, T_n\} \cup \{X_{ij}, \overline{X}_{ij}, X'_{ij}, \overline{X}'_{ij} \mid 1 \leq i \leq m \wedge 1 \leq j \leq n\}$ . Let the rules of  $\mathcal{G}'$  be as shown in Fig 1, except that the mapping  $\gamma$  is now defined as

- $\gamma_{jk} = X'_{ij}$  for  $y_{jk} = x_i$ , and
- $\gamma_{jk} = \overline{X}'_{ij}$  for  $y_{jk} = \overline{x}_i$ .

Define ( $\equiv$ ) such that

- $X_{ij} \equiv X'_{ij}$  for  $i = 1, \dots, m$  and  $j = 1, \dots, n$ ,
- $\overline{X}_{ij} \equiv \overline{X}'_{ij}$  for  $i = 1, \dots, m$  and  $j = 1, \dots, n$ , and
- no other nontrivial equivalences hold.

Observe that the grammar  $\mathcal{G}'$  doesn't have any recursion involved, so its language is finite. In fact,  $\omega = a^{(n+1) \cdot m} b(ce)^n d$  from Lem. 5 is the only word that can be derived from  $\mathcal{G}'$ , but there are lots of different derivations that accomplish this. Furthermore, the quotient grammar  $\mathcal{G}' / \equiv$  just yields the grammar  $\mathcal{G}$  from Def. 3. Each derivation from  $\mathcal{G}'$  corresponds to a derivation from  $\mathcal{G}$ , but not vice versa, as observed in Def. 13.

A derivation of some word, i.e.  $\omega$ , from  $\mathcal{G}'$  is repetition-free mod.  $\equiv$  iff that derivation, taken from  $\mathcal{G}$ , is repetition-free, that is, iff (by Lem. 5)  $\kappa$  is satisfiable.  $\square$

<sup>9</sup> This definition is used in connection with minization of deterministic finite automata, but often left implicit in textbooks (e.g. [HU79, Sect.3.4, p.65–71]); see e.g. [GJ07, p.5] for an explicit definition.

Cor. 14 subduces our hope to find an efficient algorithm to decide whether a law term (constructed from a given set of operators) for a given sequence  $s$  is guessable from a given prefix  $s'$ .

Note, however, that repetition-free derivability  $\text{mod. } \equiv$  is a necessary, but not sufficient condition for  $t$  being minimal w.r.t. some notion of *size*. There are repetition-free ( $\text{mod. } \equiv$ ) derivable terms that are nevertheless non-minimal w.r.t. every reasonable notion of *size*. For example,  $v_p + v_1$  is a construction law term for the sequence 1; 2, 4, 7, none of its subterms is a law for its proper prefix 1; 2, 4,<sup>10</sup> yet every admitted definition of a *size* notion will either make  $v_1 + v_1$  a smaller or equal term, or  $v_p + v_p$ , both are laws for 1; 2, 4.

As a consequence, the above guessability task could still be efficiently decidable.

## References

- [AHU74] Alfred V. Aho, John E. Hopcroft, and Jeffrey D. Ullman. *The Design and Analysis of Computer Algorithms*. Addison-Wesley, 1974.
- [Bur05] Jochen Burghardt. *E-generalization using grammars*. *Artificial Intelligence Journal*, 165(1):1–35, 2005.
- [CDG<sup>+</sup>08] H. Comon, M. Dauchet, R. Gilleron, F. Jacquemard, D. Lugiez, S. Tison, and M. Tommasi. *Tree Automata Techniques and Applications (TATA)*. <https://gforge.inria.fr/frs/download.php/10994/tata.pdf>, Oct 2008.
- [GJ07] Tristan le Gall and Bertrand Jeannet. Analysis of communicating infinite state machines using lattice automata. Publication Interne 1839, Institut de Recherche en Informatique et Systèmes Aléatoires (IRISA) — Campus Universitaire de Beaulieu, Mar 2007.
- [Hei95] Birgit Heinz. *Anti-Unifikation modulo Gleichungstheorie und deren Anwendung zur Lemmagenerierung*. PhD thesis, TU Berlin, Dec 1995.
- [HMu03] John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison Wesley, Upper Saddle River/NJ, 2003.
- [HU79] John E. Hopcroft and Jeffrey D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley, Reading/MA, 1979.
- [Sip97] Michael Sipser. *Introduction to the Theory of Computation*. PWS Publishing Co., Boston/MA, 1997.

---

<sup>10</sup> i.e. the term  $v_p + v_1$  has a repetition-free derivation  $\text{mod. } \equiv$ , where factorizing by the latter turns the grammar for 1; 2, 4, 7 into that for 1; 2, 4